Search

Objective

As projects grow with numerous files, there is a frequent need to search the files for words/ patterns. This projectlet attempts to provide a tool to meet the need.

User needs and requirements

Id	Need/Requirement
1	The utility shall support the search for text or regular expressions.
2	The utility shall support the search of specified files.
3	The utility shall support the search of all files in specified directories.
4	The utility shall support recursive search of files in specified directories.
5	The occurrences shall be listed with line numbers.
6	File names shall be listed only if the search candidate is found in the files.

User wants

The following will enhance the appeal of the tool.

Id	Wants
1	For search patterns, the actual text that matched shall be shown distinctively
2	All instances of the candidate shall be identified in each line

Specifications

Switch	Interpretation
-p | —pattern	Search for the pattern instead of as text
-r | —recurse	Recursive search
-c | —case	Case sensitive search. Default is non case sensitive
arg0	Search candidate (pattern or text)
arg1 … argn	Files to search. File names are always assumed to be wildcards.

Switch	Interpretation
	If the name is a directory, then all the files in the directory are searched
	Directories are searched recursively if the recurse option is specified.

Modular structure

This projectlet is an opportunity to explore ideas on long term maintainability of applications. Separating for example the command line processing and the implementation of the features will yield results in the long run. This example could potentially be expanded to provide a GUI instead of the command line - in which case the implementation could / should be reused.

Example usage

SEARCH FOR STRING IN FILES (*.CS)

```
bin/Debug/search.exe -r:. Impl "*.cs"
------------------------------------------------------------------
File ./Impl.cs
------------------------------------------------------------------
00007:     public class Impl
00144:         public Impl(Cli cli)
------------------------------------------------------------------
File ./Search.cs
------------------------------------------------------------------
00010:             Impl impl = new Impl(cli);
```

SEARCH FOR PATTERN IN FILES (*.CS)

```
bin/Debug/search.exe -r:. -p "\{.*\}"  "*.cs"
------------------------------------------------------------------
File ./Properties/AssemblyInfo.cs
------------------------------------------------------------------
00012: [assembly: AssemblyCopyright("${AuthorCopyright}")]
00016: // The assembly version has the format "{Major}.{Minor}.
{Build}.{Revision}".
00017: // The form "{Major}.{Minor}.*" will automatically update the
build and revision,
00018: // and "{Major}.{Minor}.{Build}.*" will update just the
revision.
------------------------------------------------------------------
File ./Cli.cs
------------------------------------------------------------------
00020:         public List<string> Arguments { get => arguments; }
00024:             Console.WriteLine($"Pattern={pattern} Case
sensitive={caseSensitive} recursive={recursive} toplevel={toplevel}");
00025:             Console.WriteLine($"Candidate={candidate}");
00033:             Console.WriteLine($"-{sw}\t| --{longsw} \t-
{help}");
00038:             Console.WriteLine($"{NAME} - {MAJOR_VERSION}.
{MINOR_VERSION}");
```

```
00062:             Console.WriteLine($"Unrecognized switch {arg}");
-------------------------------------------------------------------
File ./Impl.cs
-------------------------------------------------------------------
00017:             Console.WriteLine($"File {filename}");
00027:                 Console.WriteLine($"Size {info.Length}");
00037:                 Console.WriteLine($"RecursiveSearch {dirname}
fname={fname}");
00073:                 Console.WriteLine($"{dirname} is not a
directory. No recursive search possible");
00083:                 Console.WriteLine($"Searching {filename}");
00090:                     Console.WriteLine($"{filename} is a
diretory. skipping");
00102:                 Console.WriteLine($"Error opening {filename}");
00121:                         Console.WriteLine($": {line}");
00135:                         Console.WriteLine($": {line}");
```